Analyse financière :
Cours de l'action Apple

Mohammed Bouri

Introduction

Dans le cadre du cours « Marchés financiers et gestion de portefeuille », le présent rapport a pour objet de présenter une analyse financière d'une série temporelle, en particulier celle des « prix » d'APPLE.

Figure 1: Logo d'APPLE

Pour cela j'ai fait une recherche dans la plateforme Yahoo, en particulier « Yahoo-Finance », pour avoir une base de données qui me permettra d'avoir une idée générale sur le comportement des variables à savoir : « Open, High, Low, Close, Adj.Close, Volume » selon la variable « Date ».

Objectifs, Matériels et Outils

Objectifs :

L'analyse financière est un ensemble de méthodes, de démarches et d'outils qui permet d'étudier et de juger la situation et les résultats financiers d'une entreprise ou d'un ensemble d'entreprises.

C'est dans ce contexte que mon travail peut être ancré. Il a pour but essentiel de répondre, au niveau de divers volets, à un certain nombre d'objectifs que j'ai préalablement fixés, à savoir :

- Présentation et analyse des données (Statistique descriptive)
- Estimations et Tests (Inférence, Séries temporelles) : Rappels théoriques, estimateurs, lois, lecture des sorties R.
- Validation des propriétés statistiques (propriétés des données financières).
- Modélisation et prédiction à l'aide des séries temporelles.
- Modélisation et prédiction à l'aide des techniques de lissage. . .

Matériels et outils :

Lors du traitement des données, j'ai opté pour le langage R, qui est considéré comme étant le programme le plus performant en matière de statistique. Ce choix est justifié par plusieurs raisons :

- Le langage R est un langage Open source, facilite la manipulation des petites bases de données comme la mienne, et génère des graphes représentatifs (Tous les graphes du rapport sont des sorties du langage R)

Le processus de l'étude

Etape 0 :

Installation des packages :

Quantmod: Specify, build, trade, and analyse quantitative financial trading strategies.

Forecast: Methods and tools for displaying and analysing univariate time series forecasts including exponential smoothing via state space models and automatic ARIMA modelling.

Tseries: Time series analysis and computational finance.

TimeSeries: Basic functions such as scaling and sorting, sub setting, mathematical operations and statistical functions.

Xts: Provide for uniform handling of R's different time-based data classes by extending zoo, maximizing native format information preservation and allowing for user level customization and extension, while simplifying cross-class interoperability.

Importation des données :

J'ai importé ma base de données du site Yahoo-Finance grâce à la commande :

```
#Importer les données d'un fichier déjà télécharger depuis "Yahoo Fianance"
data=getSymbols(c('AAPL'),src='yahoo', from='2015-04-01', to='2020-04-01',auto.assign=FALSE)
```

Visualisation des données :

	AAPL.Open	AAPL.High	AAPL.Low	AAPL.Close	AAPL.Volume	AAPL.Adjusted
2015-04-01	124.82	125.12	123.10	124.25	40621400	114.34224
2015-04-02	125.03	125.56	124.19	125.32	32220100	115.32691
2015-04-06	124.47	127.51	124.33	127.35	37194000	117.19503
2015-04-07	127.64	128.12	125.98	126.01	35012300	115.96191
2015-04-08	125.85	126.40	124.97	125.60	37329200	115.58459
2015-04-09	125.85	126.58	124.66	126.56	32484000	116.46805
2015-04-10	125.95	127.21	125.26	127.10	40188000	116.96499
2015-04-13	128.37	128.57	126.61	126.85	36365100	116.73492
2015-04-14	127.00	127.29	125.91	126.30	25524600	116.22879
2015-04-15	126.41	127.13	126.01	126.78	28970400	116.67050
2015-04-16	126.28	127.10	126.11	126.17	28369000	116.10913
2015-04-17	125.55	126.14	124.46	124.75	51957000	114.80237
2015-04-20	125.57	128.12	125.17	127.60	47054300	117.42512
2015-04-21	128.10	128.20	126.67	126.91	32435100	116.79014
2015-04-22	126.99	128.87	126.32	128.62	37654500	118.36378
2015-04-23	128.30	130.42	128.14	129.67	45770900	119.33005
2015-04-24	130.49	130.63	129.23	130.28	44525900	119.89141

Figure 2: Les 17 lignes de la variable "data"

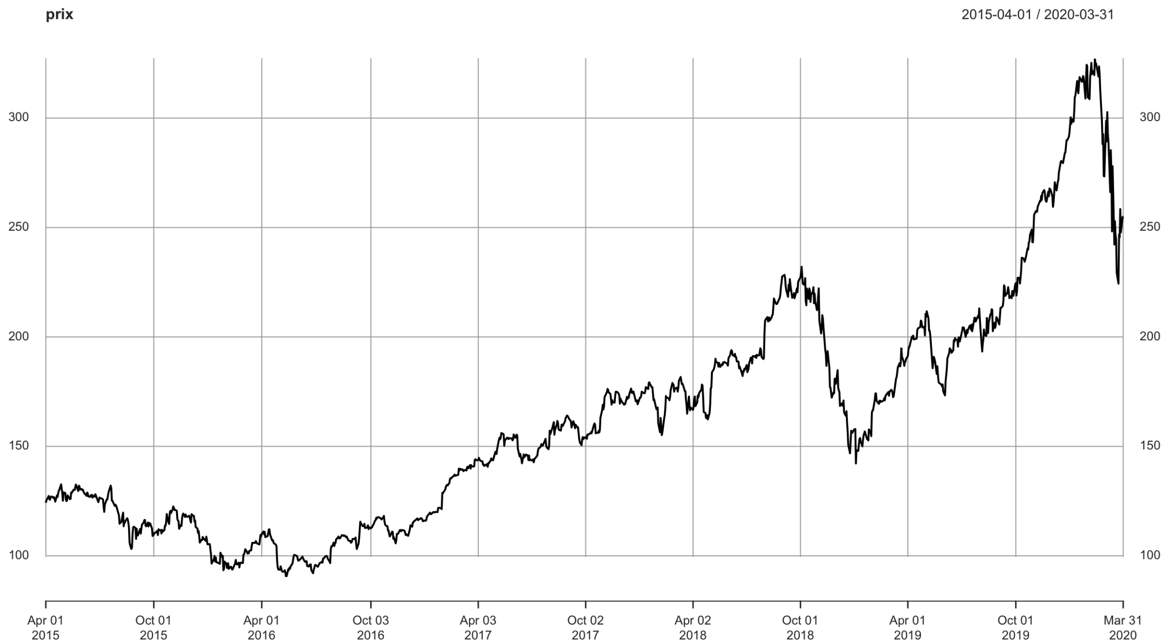

Figure 3: Les prix durant 5 ans

Modèle utilisé :

Pour modéliser mon étude, j'ai adopté le modèle « ARIMA » qui permet de représenter sous une forme succincte certains phénomènes variant avec le temps, et de faire des prévisions pour les valeurs futures du phénomène, avec un intervalle de confiance autour des prévisions.

Alors le but est d'ajuster mes données avec la notation compatible au modèle ARIMA puis le valider.

Etape 1 : Décomposer les données

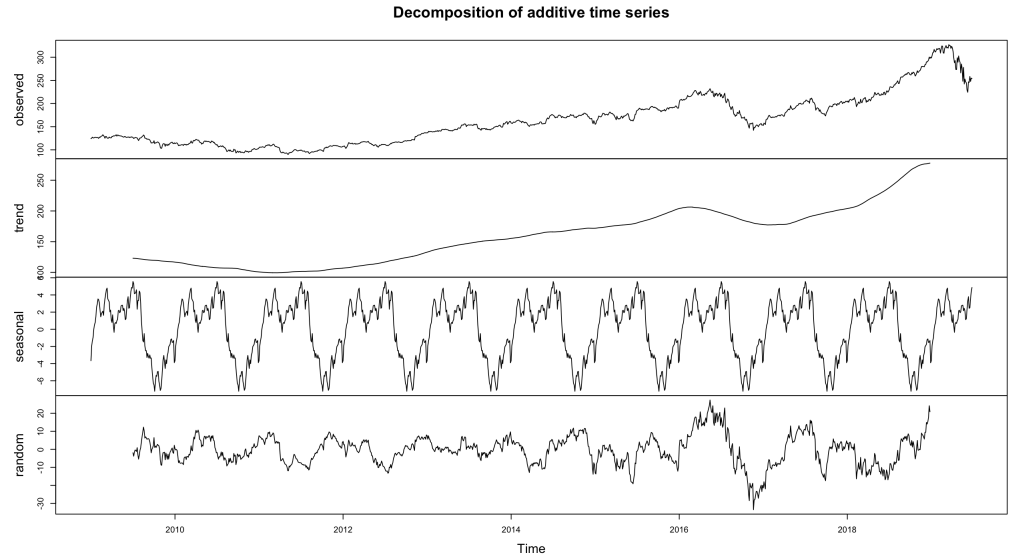

Figure 4: Décomposition de la série

Les données observées : En examinant brièvement les données, nous pouvons constater que la valeur des actions a augmenté au fil du temps depuis 2016, mais a augmenté à la fin de 2018. Il n'y a pas de modèle apparent qui peut être utilisé pour mettre à l'échelle la valeur du prix des actions vu que les tendances ne sont pas linéaires, et il n'y a pas de formule mathématique pour décrire le changement de courbe ou de fluctuations entre l'augmentation ou la baisse des prix, alors que le cours des actions baisse au début de 2019.

Tendance : décrit le schéma général de la série sur toute la période, en tenant compte des augmentations et des diminutions des prix ensemble, on remarque qu'elle est globalement à la hausse.

La composante saisonnière : décrit les fluctuations du cours des actions en fonction de l'exercice. D'après le graphique ci-dessus, le pic du prix des actions se produit chaque année au quatrième trimestre (octobre, novembre et décembre) et le creux du prix des actions se produit chaque année au premier trimestre (janvier, février et mars), avec des fluctuations oscillantes claires entre les deux.

L'erreur (Résidu, Bruit) : décrit les tendances qui ne peuvent pas être expliquées par une tendance ou par des composantes saisonnières.
Statistiquement, ces erreurs sont la différence entre le prix observé et le prix estimé. L'erreur aléatoire est particulièrement importante pour ce projet car un modèle statistique ne peut être ajusté que si ***les résidus sont indépendants et distribués indépendamment.***

Remarque :

D'après cette analyse, on peut remarquer qu'il y a une tendance claire que nous pouvons observer à partir de chacune des composantes de données. Alors pour qu'on puisse tenir compte des résidus, il vaut mieux reconfigurer les données de telle façon de diminuer le nombre des points aberrants dans le but de satisfaire les conditions du modèle ARIMA.

Etape 2 : Reconfigurer des données

Pour stabiliser la variance, je vais extraire les rendements logarithmiques des prix.

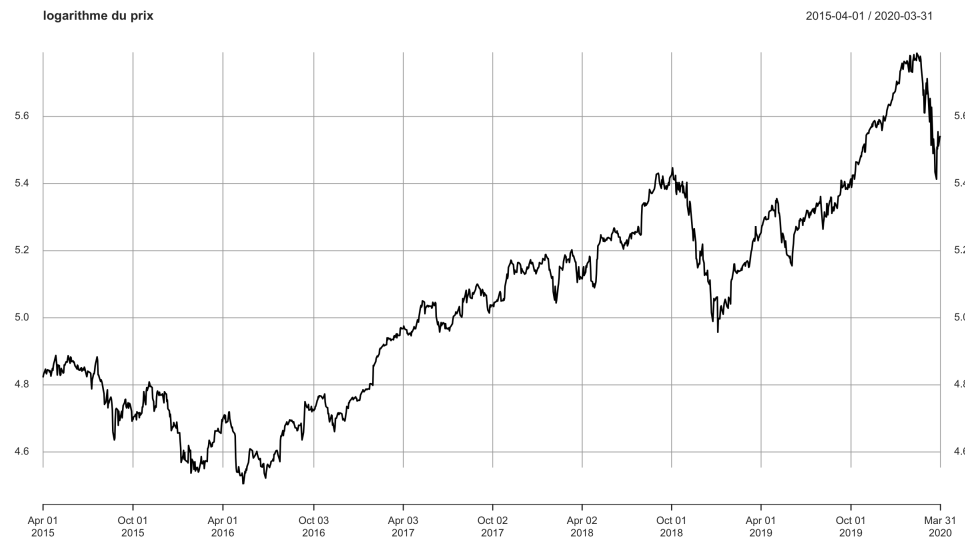

Figure 5: Logarithme des Prix

Lors de la visualisation des graphiques, la plage de valeurs de l'axe des y est beaucoup plus petite que la plage de valeurs dans les données observées. Par rapport aux fluctuations observées dans les données brutes de l'étape 1, les tracés de données transformés sont plus linéaires.

Pour lisser encore plus les données, il faut supprimer complètement les fluctuations des données. Je différencie les retours logarithmiques pour transformer les données dans leurs formes stationnaires.

La stationnarité des données est utile pour les modèles de prévision, car un analyste peut simplement conclure à partir de la prédiction que les tendances d'un graphique seront les mêmes à l'avenir que par le passé.

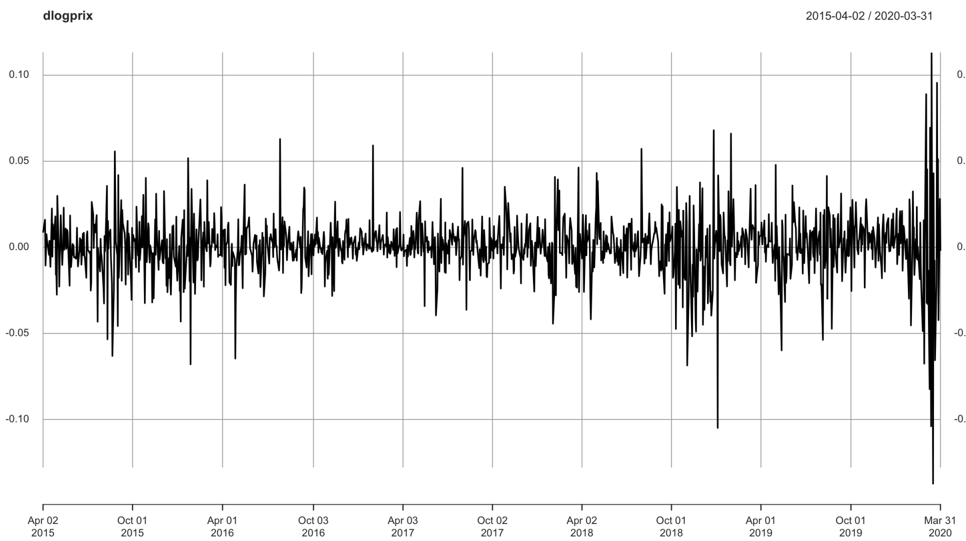

Figure 6: Différentielle du logarithme des prix

La différentielle du premier ordre fait ce qui suit :

- Les propriétés statistiques, telles que la moyenne, la variance et l'autocorrélation, sont constantes dans le temps.
- Les propriétés linéaires, telles que l'ordonnée à l'origine et la pente, sont constantes dans le temps.

Remarque :
Les données transformées oscillent autour de 0. Les écarts par rapport à 0 sont beaucoup plus faibles que dans les graphiques précédents, ce qui signifie que les données ont été plus lissées.

Etape 3 : Le choix des données

Avant d'adapter un modèle ARIMA, il faut confirmer que les données sont réellement utilisables, c'est-à-dire confirmer la stationnarité.
Pour cela, le test Dickey Fuller détermine si une racine unitaire, une caractéristique pouvant entraîner des problèmes d'inférence statistique, sous les hypothèses :

- **H0 : Les données de notre série chronologique incluent une racine unitaire et ne sont pas stationnaires.**
- **H1 : Les données de notre série chronologique n'incluent aucune racine unitaire et elles sont stationnaires**

```
        Augmented Dickey-Fuller Test

data:  logprix
Dickey-Fuller = -2.8207, Lag order = 10, p-value = 0.2309
alternative hypothesis: stationary
```

Figure 7: Sortie du test sur le log(prix)

```
        Augmented Dickey-Fuller Test

data:  dlogprix
Dickey-Fuller = -9.8484, Lag order = 10, p-value = 0.01
alternative hypothesis: stationary

Warning message:
In adf.test(dlogprix) : p-value smaller than printed p-value
```

Figure 8: Sortie du test sur le dlog(prix)

Alors on conclut qu'on va utiliser les données du « **dlogprix** » car :

$$p - value < 5\% \implies \text{On rejette H0}$$
$$\implies \text{La série est stationnaire}$$

Le modèle ARIMA (p,d,q)

Etape 4 : Savoir les paramètres du modèle

Pour savoir les paramètres p et q, il faut passer par les fonctions d'autocorrélations pour déterminer q et par les fonctions d'autocorrélations partielles pour déterminer p.

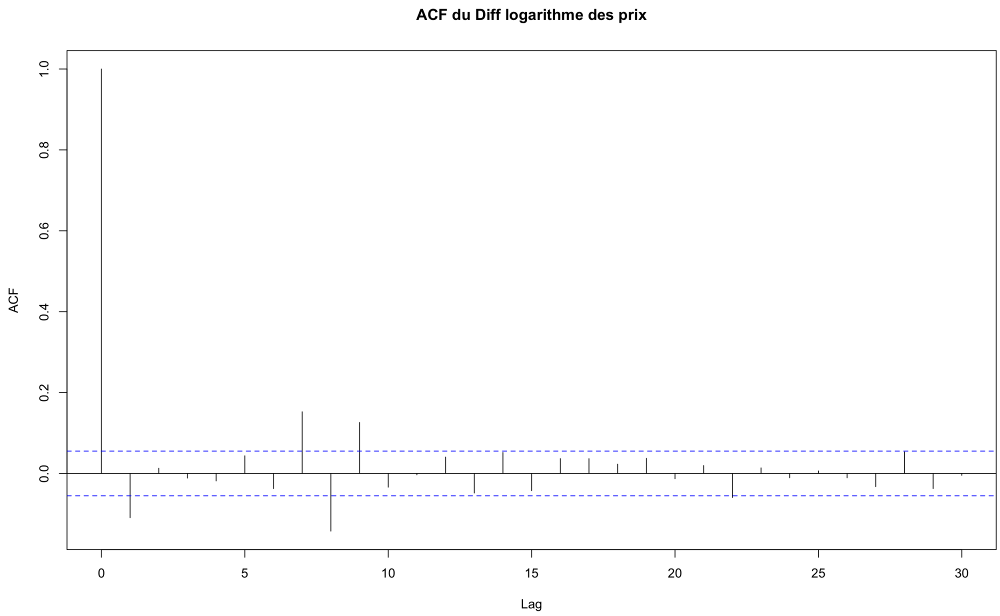

Figure 9: Fonction d'autocorrélation du "dlogprix"

On voit où les valeurs ne traversent plus la ligne pointillée bleue, donc $\boxed{\mathbf{p = 1}}$

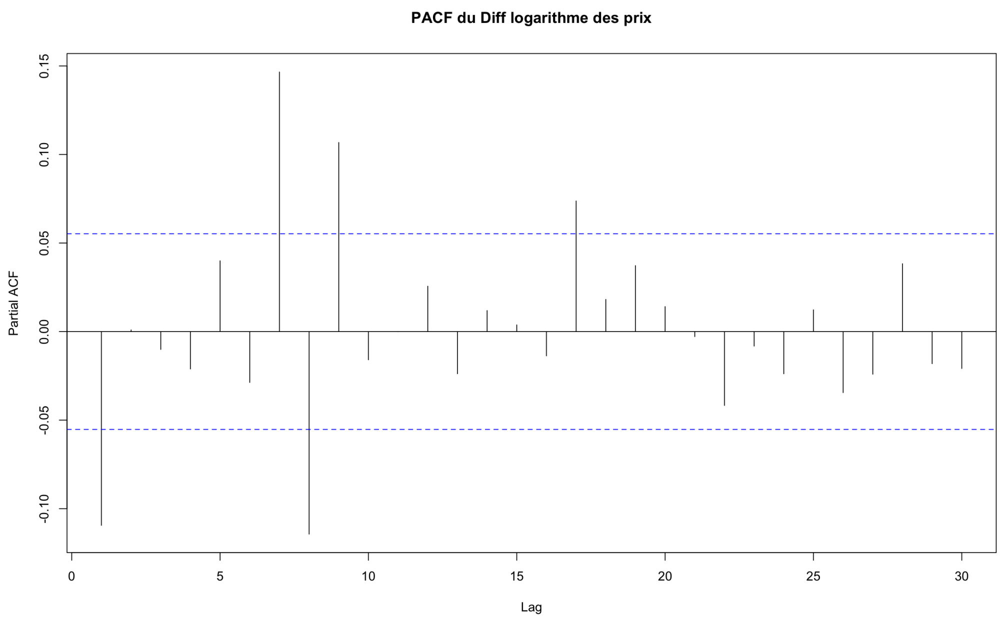

Figure 10: Fonction d'autocorrélation partielle du "dlogprix"

On voit où les valeurs ne traversent plus la ligne pointillée bleue, donc $\boxed{\mathbf{q = 17}}$

Programmation d'une prévision ajustée : Avec le modèle ARIMA (1,0,17) (D'après ma visualisation d'ACF & PACF)

Maintenant que nous en sommes à ajuster un modèle ARIMA (1,0,17) pour les rendements de prix logarithmiques différentiels, nous pouvons construire une prévision. Pour commencer la prévision, j'ai fait une comparaison entre les rendements réels et les rendements prévus. Pour faire cela, j'ai créé deux nouvelles variables :

- **reelreturn :** sous format « xls » pour que je puisse manipuler les données comme une fonction de temps.
- **forcastreturn :** sous format « data.frame » pour que je puisse insérer les résultats du modèle de prévision.

Je vais diviser l'ensemble de données en :

- Ensemble de formation « dlogprix_training »
- Ensemble de test « dlogprix_testing » en introduisant un point d'arrêt.

tel que :

- **« dlogprix_training »** est utilisé pour ajuster les paramètres fixés par le modèle ARIMA (1,0,17) pour les estimations.
- **« dlogprix_testing »** est utilisé pour évaluer les performances du modèle en fournissant une évaluation non biais de son ajustement.

En d'autres termes, l'ensemble de formation est la base de nos résultats et l'ensemble de test est l'endroit où nous déterminons les prévisions.

Après, j'ai utilisé le test de « Box-Ljung » qui teste le caractère aléatoire global, ou le manque d'ajustement, de chaque résidu sur un groupe de retards dans les données sous les hypothèses :

- **H0 : Les données sont distribuées de manière indépendante, relativement aléatoires et devraient s'intégrer correctement dans le modèle.**
- **H1 : Les données ne sont pas distribuées de manière indépendante, relativement aléatoires et ne devraient pas s'intégrer correctement dans le modèle.**

Le but de ce test est de choisir un modèle qui a des résidus indépendants et distribués indépendamment.

Remarque :
Pour chaque point de données qui ne réussit pas le test Box-Ljung, la boucle émet un avertissement. Le nombre d'avertissements permet de renforcer notre interprétation du modèle et de confirmer la validité des tendances.

```
Forecast method: ARIMA(1,0,17) with zero mean

Model Information:

Call:
arima(x = dlogprix_training, order = c(1, 0, 17), include.mean = FALSE)

Coefficients:
         ar1      ma1     ma2     ma3      ma4     ma5      ma6     ma7      ma8     ma9     ma10    ma11    ma12
      0.5299  -0.6028  0.0171  0.0098  -0.0101  0.0468  -0.0365  0.1491  -0.2083  0.1977  -0.0993  0.0105  0.0419
s.e.  0.1971   0.1978  0.0368  0.0334   0.0329  0.0330   0.0339  0.0336   0.0448  0.0439   0.0431  0.0352  0.0347
        ma13    ma14     ma15    ma16    ma17
     -0.0544  0.0637  -0.0546  0.0515  0.0417
s.e.   0.0355  0.0324   0.0341  0.0354  0.0367

sigma^2 estimated as 0.0002953:  log likelihood = 3324.27,  aic = -6610.54

Error measures:
                       ME       RMSE        MAE MPE MAPE      MASE          ACF1
Training set 0.0004689313 0.01718391 0.01179325 NaN  Inf 0.6811767 -0.0001679404

Forecasts:
     Point Forecast      Lo 99      Hi 99
1258   -0.006107787 -0.0503706 0.03815503
```

Figure 11: Les coefficients du modèle

Alors le modèle s'écrit sous la forme :

$$\boxed{\begin{gathered} Y_t = 0.5299Y_{t-1} - 0.6028\varepsilon_{t-1} + 0.0171\varepsilon_{t-2} + 0.0098\varepsilon_{t-3} - 0.0101\varepsilon_{t-4} + 0.0468\varepsilon_{t-5} - \\ 0.0365\varepsilon_{t-6} + 0.1491\varepsilon_{t-7} - 0.2083\varepsilon_{t-8} + 0.1977\varepsilon_{t-9} - 0.0993\varepsilon_{t-10} + 0.0105\varepsilon_{t-11} + \\ 0.0419\varepsilon_{t-12} - 0.0544\varepsilon_{t-13} + 0.0637\varepsilon_{t-14} - 0.0546\varepsilon_{t-15} + 0.0515\varepsilon_{t-16} + 0.0417\varepsilon_{t-17} \end{gathered}}$$

Visualisation et validation du modèle

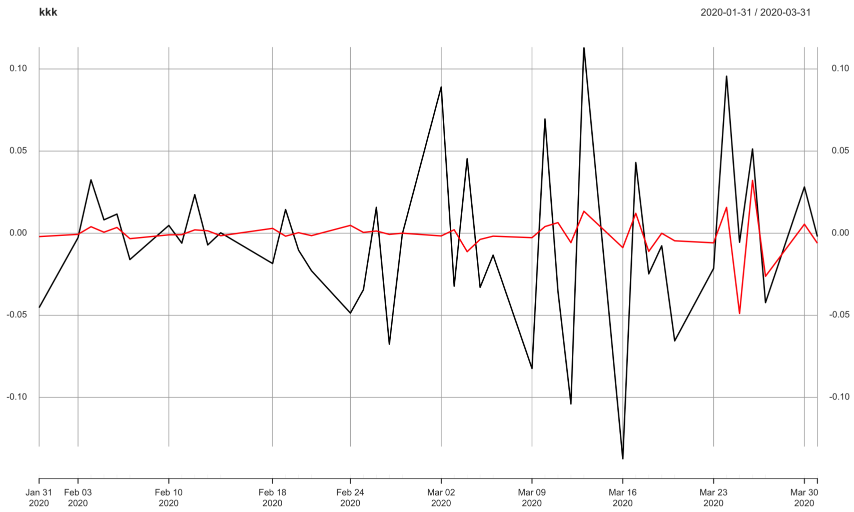

Figure 12: Actual returns (Black) Vs Forecast returns (Red)

En observant la ligne rouge, nous pouvons remarquer que les augmentations se produisent légèrement avant que les tendances ne se produisent dans la ligne noire.

Visualisation de la prévision

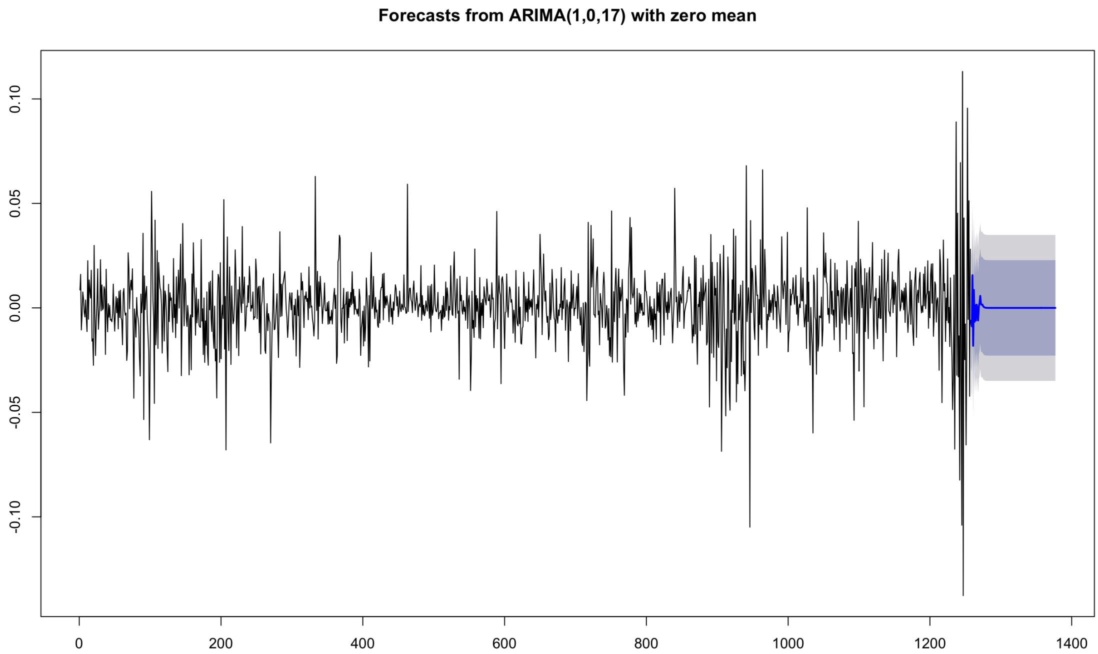

Figure 13: Prévision du "dlogprix"

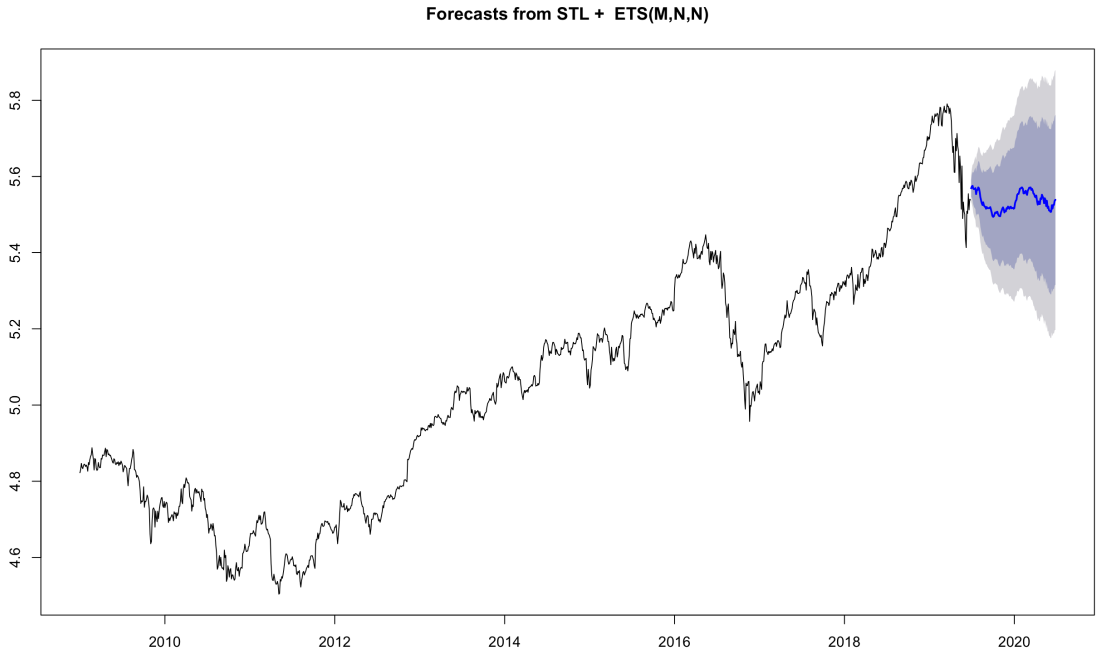

Figure 14: Prévision du logarithme des données observées

J'ai fusionné les séries de retours réels et prévisionnels des jeux de tests et j'ai créé une comparaison binaire ainsi j'ai calculé un pourcentage de précision.

```
> print(comparaison)
             realreturn     Forecasted accuracy
2020-01-31 -0.0453517512 -2.074415e-03        1
2020-02-03 -0.0027500737 -7.036436e-04        1
2020-02-04  0.0324804311  3.951856e-03        1
2020-02-05  0.0081212563  5.958847e-04        1
2020-02-06  0.0116290519  3.490557e-03        1
2020-02-07 -0.0160563616 -3.336971e-03        1
2020-02-10  0.0047382770 -1.004361e-03        0
2020-02-11 -0.0060515597 -8.984058e-04        1
2020-02-12  0.0234701858  2.026826e-03        1
2020-02-13 -0.0071465545  1.462994e-03        0
2020-02-14  0.0002462744 -1.559020e-03        0
2020-02-18 -0.0184802585  2.950679e-03        0
2020-02-19  0.0143788697 -1.819217e-03        0
2020-02-20 -0.0103119534  3.106351e-04        0
2020-02-21 -0.0228951353 -1.489299e-03        1
2020-02-24 -0.0486655763  4.774738e-03        0
2020-02-25 -0.0344591333  4.743822e-04        0
2020-02-26  0.0157391610  1.323872e-03        1
2020-02-27 -0.0676026301 -7.267509e-04        1
```

Figure 15: Extrait du résultat

Remarque :

La précision binaire renvoie 1 si les signes calculés à partir du programme correspondent et 0 si les signes calculés à partir du programme ne correspondent pas. Le pourcentage de précision est le nombre de 1 divisé par le nombre de points de données.

Après le calcul du pourcentage :

```
> print(accuracy_pourcentage)
[1] 69.04762
```

Figure 16: Pourcentage de précision

Nous savons maintenant que le modèle est précis à 69% lors de la prévision d'une augmentation ou d'une diminution du rendement logarithmique d'un stock.

Conclusion

La valeur des actions AAPL a augmenté au fil du temps et si la tendance se poursuit, elle devrait continuer de croître. Selon le modèle, les valeurs devraient augmenter, car les rendements prévus sont plus souvent supérieurs à 0 qu'en dessous.

Le modèle construit ici n'est pas assez solide pour être utilisé pour des stratégies d'investissement, mais il peut nous dire avec précision si la valeur des actions augmentera ou diminuera à court terme

Références

https://www.datascience.com/blog/introduction-to-forecasting-with-arima-in-r-learn-data-science-tutorials
http://www.forecastingsolutions.com/arima.html
https://www.youtube.com/watch?v=N_XKJqr-VT4&t=358s
https://en.wikipedia.org/wiki/Autoregressive_integrated_moving_average
https://towardsdatascience.com/an-end-to-end-project-on-time-series-analysis-and-forecasting-with-python-4835e6bf050b
https://datascienceplus.com/time-series-analysis-using-arima-model-in-r/